

## Le protocole réseau IPv4 – Sommaire

IP – Objectifs et fonctions

Format d'un datagramme IPv4

Format d'un datagramme IPv4

Champs de l'en-tête IPv4

Fragmentation d'un datagramme IPv4

Fragmentation d'un datagramme IPv4

Sous-réseaux

Découper en sous-réseaux

Conclusion

Besoin d'un nouveau protocole ?

## IP – Objectifs et fonctions

Protocole de niveau paquet (réseau) actuellement utilisé sur l'Internet ; majoritairement dans sa version IPv4 mais bientôt remplacée par IPv6

### Interconnexion de réseaux hétérogènes

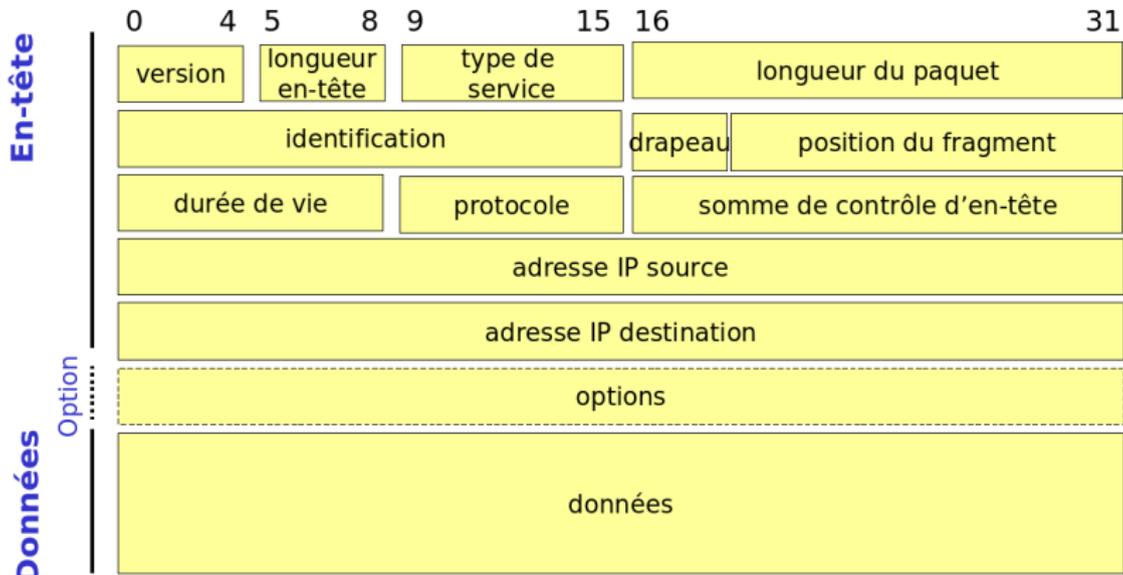
- **Adressage universel** (interconnexion de n'importe quel type d'hôte)
  - hiérarchique à deux niveaux (réseau-hôte)
  - unicité garantie par un organisme fournisseur : IANA (Internet Assigned Numbers Authority)
- Capable de **transporter plusieurs protocoles** (un champ indique le protocole transporté)
- **Adaptation au réseau physique sous-jacent** fragmentation / réassemblage selon capacités d'emport de la liaison (MTU)

## IP – Objectifs et fonctions

Qualité de Service (QoS) de type **Best effort** (au mieux)

- **Transmission non fiable : Service en mode sans connexion et sans acquittement des données**
  - indépendance des datagrammes, possibilité de perte de paquet, de duplication, de livraison dans le désordre et de congestion des routeurs
  - garantie d'acheminement gérée par couche supérieure
  - un TTL (Time To Live) pour la prévention de la congestion
- Communication **avec faible contrôle d'erreurs**
  - Si erreur détectée, tentative d'envoi d'un paquet ICMP
  - La qualité du service laissée à la charge de couche supérieure

## Format d'un datagramme IPv4



- La taille des datagrammes IP ne doit pas excéder 65536 octets
- Taille de l'en-tête = 20 octets (i.e.  $5 * 4$  octets ou  $5 * 32$  bits)

## Champs de l'en-tête IPv4

- Version : 4 bits
  - Format du protocole IP ;
  - Permet la **cohabitation de IPv4 (0100) et IPv6 (0110)** sur un même réseau
- Longueur de l'en-tête (IHL – Internet Header Length) : 4 bits
  - **En mots de 32 bits** (min 5, max 15)
  - **Permet de détecter la présence du champ Option**

- **Type de service (TOS) : 8 bits**

priorité.délai.débit.fiabilité.coût.réservé

7	6	5	4	3	2	1	0
priorité			TOS				0

- E.g. SMTP, envoi de données : 000.1.0.0.0.0 (minimise le délai)  
SMTP, contrôle/commande : 000.0.1.0.0.0 (maximise le débit)
- Défini dans RFC 791 ; depuis révisé dans RFC 1812
- **Taille en octets du datagramme (en-tête + données) : 16 bits**
  - non fixe mais limité à 65536 o.

## Champs de l'en-tête IPv4

- Identification, 16 bits
  - Attribué aléatoirement par la source
  - **Indique à quel datagramme appartient le fragment** (utilisé avec l'adresse IP source par le destinataire)
- Drapeau (Flag), 3 bits : permet de **savoir si le datagramme est fragmenté**
  - 1er bit non utilisé ;
  - 2e bit DF (**Don't Fragment**) utilisé quand l'extrémité est incapable de réassembler ; un routeur détruit un tel paquet s'il ne peut le transférer
  - 3e bit MF (**More Fragment**) utilisé pour fragments de 1 à  $n - 1$
- Position du fragment (Offset), 13 bits
  - **Position** dans datagramme d'origine **en multiple de 8 octets** ; i.e. quantité de données déjà transmises
  - Tous les fragments ont une taille multiple de 8 octets sauf le dernier

## Champs de l'en-tête IPv4

- Durée de vie (TTL – Time To Live), 8 bits
  - **Limite la présence sur le réseau**
  - Décrémentée à chaque routeur et retirée à 0
- Protocole, 8 bits

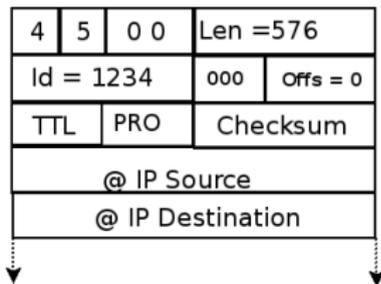
**Spécifie protocole de niveau transport à l'origine de l'émission** pour être traité par même protocole à la réception

TCP	6
UDP	16
ICMP	1

- Somme de contrôle de l'en-tête (Checksum), 16 bits
  - **Recalculée pour chaque intermédiaire** ; paquet **rejeté si erreur**
  - **Données vérifiées à la couche transport** supérieure
- **Adresses IP** source et destination, 32 bits chacune
- Options
  - Contraintes d'acheminement du datagramme
  - **Bits de bourrage** pour compléter jusqu'à 32 bits

## Fragmentation d'un datagramme IPv4

Soit un paquet IP de 576 octets à  
acheminer sur un réseau X.25  
(rappel :  $MTU(X.25) = 128$  o. dont 20o.  
en-tête + 108 o. de charge effective



- Sachant qu'un fragment doit avoir une taille de données multiple de 8, quelle taille maximale de donnée aura un fragment ?
- Combien de fragments et la taille totale de chacun (Champ Len) ?
- Que met on dans les champs position et flags de chaque paquet-fragment ?

Sachant qu'un fragment doit avoir une taille de données multiple de 8, quelle taille maximale de donnée aura un fragment ?

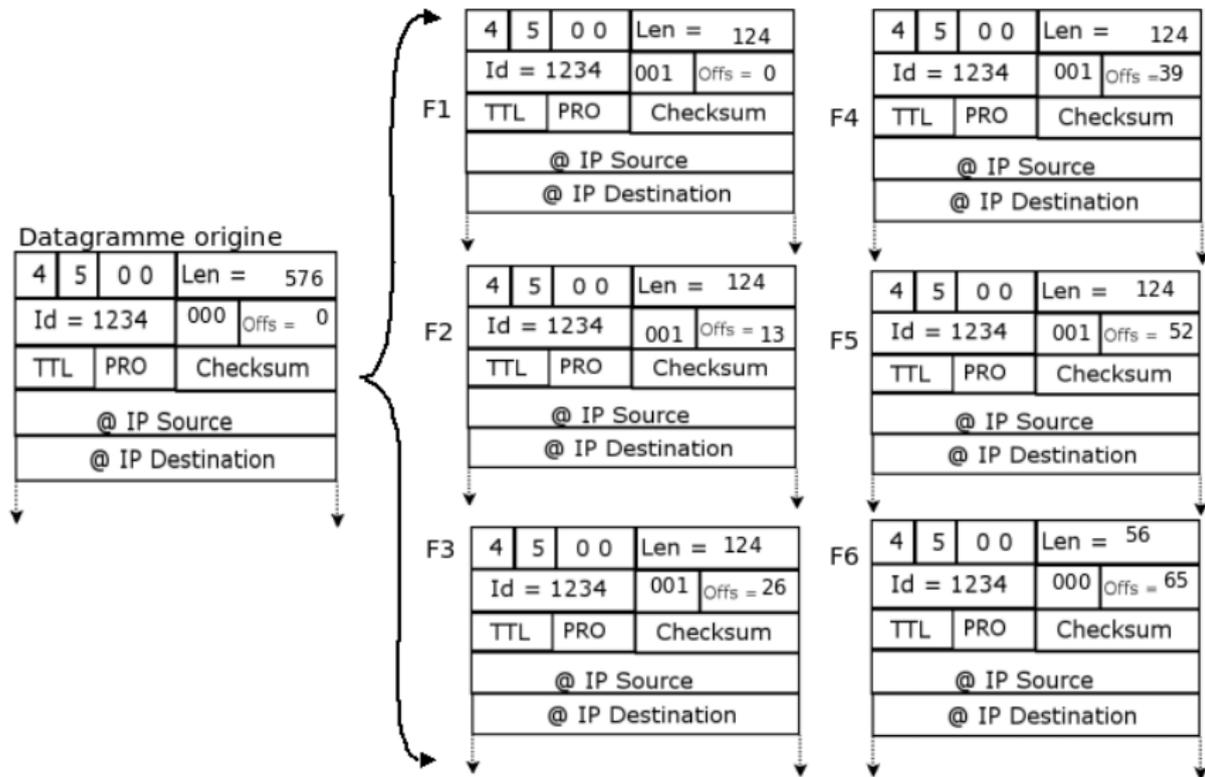
- 108 non divisible par 8 ; le plus grand nombre inférieur à 108 et divisible par 8 est 104

Combien de fragments et la taille totale de chacun (champ Len) ?

- Charge effective que l'on veut transporter est de  $(576 - 20 \text{ o. en-tête}) / 104 = 5,34$  (en fait  $5 * 104 + 36$ )
- Ce qui fait 6 paquets-fragments auxquels on rajoute à chacun les 20 o. d'en-tête
- Respectivement Longueur (Len) = 124 pour les 5 premiers Fragments et 56 pour le dernier

Que met on dans les champs position et flags de chaque paquet-fragment ?

- Les positions (Offs) sont 0 pour le 1er,  $104/8=13$  pour 2ème,  $2*(104/8) = 26$  pour 3ème ...
- Flag MF(001) sauf pour le dernier (000)



Note : Len et Offs donnés ici en décimal pour faciliter la lecture

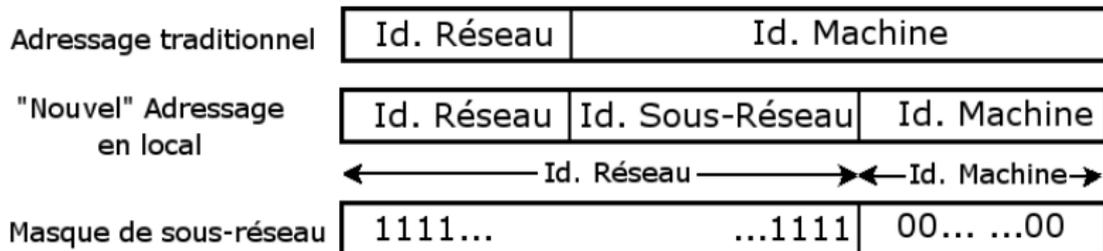
## Découper en sous-réseaux

### Problème

*Une seule adresse réseau attribuée et besoin d'en considérer plusieurs pour des raisons de sécurité, répartition du trafic, cohérence (Prof, Etudiant, etc.)*

### Solution

**Utilisation de bits de poids forts de l'identifiant machine** pour représenter des sous-réseaux



## Exemple de découpage en sous-réseaux

On souhaite découper le réseau 192.168.1.0 (Classe C) en 4 sous-réseaux

- Ad. Diffusion = 192.168.1.255
- Masque de réseau = 255.255.255.0 aussi noté 192.168.1.0/24
- Nombre d'adresses matérielles = 254 réservées (réseau et diffusion)

Pour coder 4 sous-réseaux, il faut 2 bits : 00, 01, 10, 11

- Masque de sous-réseau = 255d.255d.255d.11000000b i.e.  
255d.255d.255d.192d ou encore 192.168.1.0/26
- Adresses matérielles =  $2^6$  adresses logiques - 2 = 62
- Nombre total de matériels  $4 * 62 = 248$

Sous-réseaux	Nb. de machines	Zone d'adressage matériel	Ad. réseau	Ad. de diffusion
Sr1	62	.1 -> .62	.0	.63
Sr2	62	.65 -> .126	.64	.127
Sr3	62	.129 -> .190	.128	.191
Sr4	62	.193 -> .254	.192	.255

## Besoin d'un nouveau protocole ?

- **Epuisement des adresses IPv4 :**

"Les quatre principaux opérateurs français (Bouygues Telecom, Free, Orange, SFR) ont déjà affecté entre environ 94% et 99% des adresses IPv4 qu'ils possèdent, à fin juin 2019."<sup>1</sup>

- **NAT (traduction d'adresses privées/publique) : solution à court terme NAT mais coût de performance**
- **Explosion de la taille des tables de routage**
  - **CIDR, solution partielle qui fait disparaître les classes d'adresses**, autorise l'agrégation d'adresses de réseaux contigus en un seul préfixe, organise géographiquement l'affectation des adresses
- **Offre pauvre de services**
  - Ne prend pas en charge en natif : QoS, sécurité, mobilité

---

<sup>1</sup><https://www.arcep.fr/cartes-et-donnees/nos-publications-chiffrees/transition-ipv6/barometre-annuel-de-la-transition-vers-ipv6-en-france.html>